



# Diplomatura en Programación Web Full Stack

## Presentación

La Diplomatura Universitaria en Programación Web Full Stack acompaña a estudiantes desde nivel inicial hasta un dominio avanzado en el desarrollo de aplicaciones y sitios web dinámicos. El mercado laboral demanda desarrolladores Full Stack capaces de crear interfaces modernas (Frontend) y programar servidores y bases de datos (Backend). Por eso, la formación combina herramientas actuales con metodologías profesionales que aseguran calidad y despliegue eficiente.

Los estudiantes aprenden a programar desde cero, construyendo proyectos reales y utilizando inteligencia artificial aplicada al diseño web como apoyo estratégico. La diplomatura integra una base sólida en programación con prácticas modernas potenciadas por IA.

La diplomatura está compuesta por dos módulos que se desarrollan de manera nivelada y progresiva:

- **Programador Web Inicial – Front End Developer**, para dominar la construcción de interfaces y la lógica de programación.
- **BackEnd Developer**, para profundizar en servidores, bases de datos y seguridad.

Así, los estudiantes se preparan para convertirse en programadores web full stack y ampliar su perfil hacia el de Programador IA, con una formación completa para enfrentar los desafíos del desarrollo digital contemporáneo.



60 clases de 3 hs  
180 hs totales



Clases 100% online  
2 veces por semana



## Requisitos previos

### MANEJO GENERAL DE PC Y ARCHIVOS

El alumno debe saber utilizar una computadora y saber descargar archivos de internet.

### MANEJO DE WINDOWS Y HERRAMIENTAS OFFICE O GOOGLE WORKSPACE

Conocimientos básicos del sistema operativo Windows, así como nociones de Microsoft Word, Excel o sus equivalentes en Google Workspace.

### UTILIZACIÓN DE NAVEGADORES WEB Y USO DE ZOOM





Manejo cotidiano de navegadores como Mozilla Firefox y Google Chrome, y conocimientos básicos de Zoom para clases virtuales.

### NO SE REQUIEREN CONOCIMIENTOS O TÍTULOS PREVIOS EN PROGRAMACIÓN

Esta diplomatura está diseñada para estudiantes sin experiencia previa en programación.

## Temario detallado

### Módulo 1: Programador Web Inicial – Front End Developer

<b>Clase 1</b> 	<b>Introducción al desarrollo web + Introducción a HTML (sintaxis)</b> <ul style="list-style-type: none"><li>Arquitectura cliente/servidor y protocolos web.</li><li>Funcionamiento web y alcance de la diplomatura.</li></ul>
<b>Clase 2</b> 	<b>Git y GitHub: El flujo de trabajo profesional</b> <ul style="list-style-type: none"><li>Conceptos de control de versiones y sistema distribuido.</li><li>Inicialización de repositorios, áreas de trabajo y commits.</li></ul>
<b>Clase 3</b> 	<b>Git Avanzado: Ramas y Colaboración</b> <ul style="list-style-type: none"><li>Gestión de ramas (Branches), Merge y resolución de conflictos.</li><li>Trabajo colaborativo y publicación de sitios en GitHub Pages.</li></ul>
<b>Clase 4</b> 	<b>HTML Semántico + Formularios + Multimedia</b> <ul style="list-style-type: none"><li>Organización de la información: header, main, footer, section, article.</li><li>Entradas de datos (Inputs) y validaciones nativas.</li><li>Implementación de audio, video y figuras.</li></ul>



<b>Clase 5</b> 	<b>Introducción a CSS: Estilo y Cascada</b> <ul style="list-style-type: none"><li>• Selectores, colores y tipografías.</li><li>• La cascada, especificidad y herencia.</li></ul>
<b>Clase 6</b> 	<b>Box Model y Layouts</b> <ul style="list-style-type: none"><li>• Box Model y Layouts</li><li>• Propiedades de visualización (Display): Block vs Inline.</li></ul>
<b>Clase 7</b> 	<b>CSS Flexbox</b> <ul style="list-style-type: none"><li>• Alineación y distribución dinámica del espacio.</li><li>• Contenedores e ítems modernos sin dependencia de grillas complejas.</li></ul>
<b>Clase 8</b> 	<b>Responsive Design y Unidades Relativas</b> <ul style="list-style-type: none"><li>• Mobile First: Filosofía de diseño y breakpoints.</li><li>• Media Queries y adaptación multiplataforma.</li></ul>
<b>Clase 9</b> 	<b>CSS Animaciones y Transformaciones</b> <ul style="list-style-type: none"><li>• Transformaciones y transiciones de estado.</li><li>• Keyframes y animaciones personalizadas para mejorar la UX.</li></ul>
<b>Clase 10</b> 	<b>Práctica Integradora de Maquetación</b> <ul style="list-style-type: none"><li>• Desarrollo de una landing page completa desde cero integrando HTML5 y CSS3.</li></ul>
<b>Clase 11</b> 	<b>Introducción a CSS Frameworks</b> <ul style="list-style-type: none"><li>• Aceleración del maquetado con Tailwind CSS o Bootstrap.</li></ul>
<b>Clase 12</b> 	<b>Fundamentos de Programación con JS</b> <ul style="list-style-type: none"><li>• Tipos de datos primitivos, variables (let/const) y operadores.</li></ul>
<b>Clase 13</b> 	<b>Control de Flujo y Estructuras de Repetición</b> <ul style="list-style-type: none"><li>• Lógica condicional: if/else y switch.</li><li>• Bucles: for, while y do/while.</li></ul>



<b>Clase 14</b> 	<b>Funciones y Métodos de Datos</b> <ul style="list-style-type: none"><li>• Declaración de funciones y expresiones (Arrow functions).</li><li>• Métodos esenciales de manipulación de Strings y Numbers.</li></ul>
<b>Clase 15</b> 	<b>Arrays y Métodos de Iteración</b> <ul style="list-style-type: none"><li>• Operaciones CRUD en arreglos.</li><li>• Iteración moderna: for...of y métodos básicos (.push, .pop, .shift, .unshift).</li></ul>
<b>Clase 16</b> 	<b>Objetos Literales y Métodos de Array Avanzados</b> <ul style="list-style-type: none"><li>• Estructuras de objetos y palabra reservada this.</li><li>• Métodos funcionales de arrays (sin callbacks complejos).</li></ul>
<b>Clase 17</b> 	<b>El DOM: Selección y Manipulación</b> <ul style="list-style-type: none"><li>• Árbol de nodos y selección de elementos (querySelector).</li><li>• Modificación dinámica de contenido e interactividad básica.</li></ul>
<b>Clase 18</b> 	<b>Eventos en el DOM</b> <ul style="list-style-type: none"><li>• Manejo de la interactividad: addEventListener.</li><li>• Captura y validación de datos en formularios.</li></ul>
<b>Clase 19</b> 	<b>JS Avanzado: Gestión de Estados y Práctica</b> <ul style="list-style-type: none"><li>• Lógica de actualización de estados en el DOM antes de la transición a React.</li></ul>
<b>Clase 20</b> 	<b>El Ecosistema React: Node, NPM y Proyectos</b> <ul style="list-style-type: none"><li>• Introducción a bibliotecas de UI y Single Page Applications (SPA).</li></ul>
<b>Clase 21</b> 	<b>Componentes y Sintaxis JSX</b> <ul style="list-style-type: none"><li>• Creación de componentes funcionales y pasaje de propiedades (Props).</li></ul>
<b>Clase 22</b> 	<b>React: Renderizado Dinámico y Estado Inicial</b> <ul style="list-style-type: none"><li>• Listas y renderizado mediante .map().</li><li>• Uso de Key props e introducción a useState.</li></ul>
<b>Clase 23</b> 	<b>React: Ciclo de Vida y Consumo de APIs</b> <ul style="list-style-type: none"><li>• Hook useEffect para efectos secundarios y peticiones asíncronas (Fetch/Axios).</li></ul>



## Clase 24



### React: Formularios Controlados

- Técnica de "Single Source of Truth" aplicada a formularios y validaciones.

## Clase 25



### Enrutamiento y Navegación (React Router)

- Rutas dinámicas y parámetros. Transformación a SPA real.

## Clase 26



### Gestión de Estado Global (Context API)

- Uso de Context para compartir información global (usuario, carrito, tema).

## Clase 27



### Contexto y Rutas: Integración de Sistemas

- Rutas protegidas y lógica de autenticación básica.

## Clase 28



### Despliegue Profesional y Hosting

- Build, optimización y despliegue continuo en Vercel.

## Clase 29



### Tutoría de Proyecto Final / IA (IDE/Antigravity)

- Resolución de dudas técnicas y revisión de avances.

## Clase 30



### Tutoría y Cierre

- Entrega final, feedback y recursos para formación continua

## Módulo 2: BackEnd Developer

## Clase 31



### Nivelación y Refuerzo de JavaScript

- Repaso profundo de lógica de programación.
- Tipos de datos complejos y métodos nativos.
- Preparación del entorno para Node.js.

## Clase 32



### Introducción a TypeScript

- ¿Qué es TypeScript? Ventajas y sintaxis.
- Transpilación y configuración del entorno.
- Tipado estático e interfaces.

## Clase 33



### Programación Orientada a Objetos (POO)

- Clases, propiedades y objetos.
- El rol del Constructor.
- Herencia de clases y abstracción.



## Clase 34



### Aplicación Práctica de TypeScript

- Implementación de lógica de negocio con tipos.
- Resolución de problemas reales utilizando POO y TS.

## Clase 35



### Introducción a Bases de Datos

- ¿Qué es una Base de Datos? Relacionales.
- El lenguaje SQL: Select, filtrados y ordenamiento.
- Gestión de Tablas, campos y registros.

## Clase 36



### Modelado de Datos y Relaciones

- Diseño de diagramas Entidad-Relación.
- Integridad referencial y normalización.
- Manipulación de datos: Insert, Update y Delete.

## Clase 37



### Clase Práctica de MySQL2 + Node.js

- Elaboración de bases de datos aplicadas a proyectos.

## Clase 38



### Conceptos No Relacionales

- Diferencias arquitectónicas con SQL.
- Casos de uso y flexibilidad de esquemas.
- MongoDB

## Clase 39



### MongoDB: Consultas Básicas + mongoose + nodeJS

- Definición de esquemas y validaciones.
- Relaciones entre documentos en Mongoose.

## Clase 40



### Express: Configuración Inicial

- Servidores
- Postman
- HTTP



## Clase 41

### Express



- Rutas
- Métodos HTTP
- Controllers
- Services

## Clase 42

### Express - Middlewares & Rate Limiter



- Middlewares
- Manejo de errores
- Rate Limiter
- Error 429

## Clase 43

### Express - Multer + Firebase



- Storage

## Clase 44

### Express - Express Validator + Schema Validation



- Express Validator
- Schema Validation

## Clase 45

### Auth y Session - JWT



- Json Web Token
- Registro
- Login
- Bearer Token
- Protección de rutas

## Clase 46

### Auth y Session - Encriptación Bcrypt 1



- Hasheo de contraseñas con bcrypt.
- Validación de email

## Clase 47

### Auth y Session - Encriptación Bcrypt 2



- Hasheo de contraseñas con bcrypt
- Validación de email con nodemailer



## Clase 48



### React

- Custom hooks
- Controlled form
- Login, Registro
- Cors

## Clase 49



### React

- Validación de email
- UseSearchParams
- Recuperación password

## Clase 50



### React CRUD

- Clase práctica CRUD integración

## Clase 51



### Firestore: Firestore

- Implementación de Firestore como servicio auxiliar de base de datos
- Gestión de entornos de desarrollo.

## Clase 52



### Supabase: Realtime

- Sincronización de datos en tiempo real con supabase

## Clase 53



### Despliegue y Seguridad del Servidor

- Configuración de puertos y variables de entorno.
- Seguridad en el servidor.
- Deploy en Vercel.

## Clase 54



### Docker: Contenedores y Orquestación

- Introducción a contenedores y Dockerización de aplicaciones.
- Configuración de entornos aislados.

## Clase 55



### Testing de Backend

- Pruebas unitarias y de integración con Jest y Supertest.
- Garantía de calidad en la lógica de servidor.



## Clase 56



### Documentación de APIs (Swagger/OpenAPI)

- Creación de documentación interactiva para desarrolladores.
- Estandarización de servicios web.

## Clase 57



### Prompt Engineering para Backend

- Herramientas de IA para la optimización de código de servidor. (IDE/Antigravity)

## Clase 58



### Tutoría de Proyecto Final: Arquitectura

- Resolución de dudas técnicas y revisión de la estructura del backend.

## Clase 59



### Tutoría de Proyecto Final: Ajustes Finales

- Optimización de base de datos y debugging avanzado.

## Clase 60



### Presentación Final de Proyectos

- Demo de la aplicación y cierre de la diplomatura.